REMARKS

This application has been carefully considered in connection with the Examiner's Final Office Action dated June 19, 2006. Reconsideration and allowance are respectfully requested in view of the following.

Summary of Rejections

Claims 1-21 were pending at the time of the Final Office Action.

Claims 1, 6-7, and 11-16 were rejected under 35 USC 102(e) as being anticipated by Chiang et al. (U.S. Patent No. 6,948,174).

Claims 2-5, 8-10, and 17-21 were rejected under 35 USC 103(a) as being unpatentable over Chiang et al. (U.S. Patent No. 6,948,174).

Summary of Response

Claims 6, 11-13, 15, and 16 were cancelled.

New Claim 22 was added.

Claims 1, 7-10, 14, and 17-19 were amended.

Claims 2-5, 9, 10, 20, and 21 remain as originally or previously presently.

Remarks and Arguments are provided below.

Summary of Claims Pending

Claims 1-5, 7-10, 14, and 17-22 are currently pending following this response.

PAGE 7/14 * RCVD AT 8/7/2006 4:41:51 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-3/0 * DNIS:2738300 * CSID:9727312289 * DURATION (mm-ss):03-54

Interview Summary

A telephone interview was conducted with Examiner Douglas Blair and Elizabeth Pham on August 2, 2006. Applicants would like to thank the Examiner for his time and consideration of this matter. Claim 1 was discussed in view of Chiang et al. (U.S. Patent No. 6,948,174). The Examiner noted that Chiang et al. did not appear to disclose a middleware brokering server as taught by the present application. The Examiner suggested that Claim 1 be amended to include the limitations of Claims 6 and 11. The Examiner stated that such an amendment would give a more complete description of the claimed invention. The Examiner also stated that amending Claim 1 as suggested would make it allowable unless a reference is found that teaches a middleware brokering server as disclosed by the present application.

The Examiner also asked that the present amendment answer the following questions:

1. <u>What is middleware and how is it described in the present specification?</u>

Applicants respectfully submit that the present specification describes middleware and its advantages as follows:

> [0002] Computer systems operating under heterogeneous platforms cannot always exchange data directly among themselves. Numerous types of commercially available products known collectively as middleware have been developed to facilitate data exchange between disparate computer systems. Instead of communicating directly with each other, computer systems can send data in their native format to the middleware. The middleware then sends the data to another system in a format understandable by the second system. Among the categories of middleware are message-oriented middleware and object request brokers.
>
> [0003] If a middleware product is not used, a heterogeneous computer system would typically need to operate in a point-to-point mode, for example using message queuing as point-to-point. Under the point-to-point approach, illustrated in Figure 1, if six separate and distinct applications **120**, **122**, **124**, **126**, **128**, and **130** are to communicate, connections **101-115** must be made between every possible pair of systems. This type of configuration is undesirable for several reasons. First, the number of connections is large. The number of

PAGE 8/14 * RCVD AT 8/7/2006 4:41:51 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-3/0 * DNIS:2738300 * CSID:9727312289 * DURATION (mm-ss):03-54

> connections required in a system of *n* components is $(n^2 - n)/2$. For example, a six component system such as that in Figure 1 requires (36 · 6)/2 or 15 connections. Second, the point-to-point mode requires tight coupling between each pair of platforms. That is, if each type of technology uses its own data format, a specifically designed adapter is needed between each pair to allow communication between the two. Third, the point-to-point approach creates vendor dependency. The adapters between platforms must meet the requirements of the manufacturers of each system. If a piece of equipment is replaced, the adapters between the new equipment and all other systems must be redesigned.
>
> [0004] The use of middleware allows computer systems to operate in a broker mode, sometimes referred to as a hub and spoke configuration. This approach, as illustrated in Figure 2, is an improvement over the point-to-point approach. In this configuration, each application **220, 222, 224, 226, 228,** and **230** communicates only with the broker **232** thereby reducing the number of connections **201–206** needed. For example, this six-application system would require only six connections, numbered **201–206**, as opposed to the fifteen needed for six applications connected in the point-to-point mode. Message-oriented middleware products operating in the publish/subscribe mode are an example of brokering middleware. Since these products can typically send and receive data in the native data formats of the applications they connect, adapters are typically not needed to convert data from the format of the applications to the format of a publish/subscribe engine serving as the brokering hub. This reduces vendor dependency and increases system flexibility over the point-to-point approach.

Accordingly, the advantages of the hub and spoke configuration using middleware, as opposed to the point-to-point mode using adapters, are that the number of connections between the applications is reduced, the dependency on the vendor is reduced, and the flexibility of the system is increased.

As stated earlier, the categories of middleware include message-oriented middleware and object request brokers. IBM's MQSeries and Sun Microsystem's Java Message Service (JMS) are examples of commercial products that act as publish/subscribe message-oriented middleware. Object request brokers are the other majority category of middleware and their functions and capabilities have been standardized by the Object Management Group (OMG) in a specification known as the Common Object Request Broker Architecture (CORBA).

2. What is a middleware brokering server and why is it needed if middleware already allows data exchange between disparate computer systems?

Although middleware facilitates data exchange between disparate computer systems, communication cannot necessarily be established directly from one of these middleware products to another. For example, applications that use MQSeries as their middleware service would typically not be able to communicate with applications using JMS as their middleware service. Likewise, CORBA applications would typically not be able to communicate with either MQSeries or JMS applications. This is because each middleware service uses its own native data format and programming syntax. For example, JMS uses MapMessage as its native format, and CORBA uses the structured event format as its native format. One way of allowing these disparate applications to communicate with one another would be to connect each of the middleware services to one another in a point-to-point mode with independent adapters between each of the disparate middleware products. However, this is not desirable because of the dependency on the vendor and the inflexibility of the system. Also, the complexity of a point-to-point configuration increases significantly as the number of middleware services increases. Therefore, the middleware broker server of the present application addresses the need for computer systems operating under disparate types of middleware to communicate with each other in an efficient, verifiable, flexible, and maintainable manner. The middleware broker server acts as an intermediary, or meta-middleware, device among different middleware systems by allowing each middleware service to send messages to the middleware broker server in its native data format and programming syntax. The middleware broker server then converts the messages into a standard format known as a structured event. The middleware broker server then converts the message from the structured event into the native format of the receiving application and sends the message to the receiving application. The standard data format allows the middleware broker server to operate in the publish/subscribe manner rather than the point-to-point mode thus minimizing the number of independent adapters needed for communication between disparate middleware products. The standard format also reduces vendor dependency. In addition, the data format used by the middleware broker server allows quality of service attributes to be added to platforms that do not currently offer those features.

3. Why is IMS Connect not a middleware service?

The Office Action has suggested that IMS Connect is a middleware service. However, Applicants respectfully submit that IMS Connect improves IMS TCP/IP access and enables easier access to IMS applications and data from the Internet. IMS Connect is an adapter translating between XML documents and IMS message data structures. That is why the HTML form of Chiang et al. had to be converted to an XML-formatted message before it was sent to IMS Connect. It is not a middleware service that allows multiple disparate applications to communicate with one another. There would be no need for the middleware that converts the HTML form to an XML-formatted message if IMS was a middleware service. It only translates from one format to another. Thus, it operates as an adapter or connector, not a middleware service.

Response to Rejections under Section 102

In the Final Office Action dated June 19, 2006, Claims 1, 6-7, and 11-16 were rejected under 35 USC § 102(e) as being anticipated by Chiang et al. (U.S. Patent No. 6,948,174).

Claims 6, 11-13, 15, and 16 have been cancelled. The rejection of these claims is traversed and is now believed to be moot.

Applicants have amended Claim 1 as suggested by the Examiner and respectfully request allowance of this claim. Specifically, Claim 1 now recites, "wherein the message is converted from a native language format of the sending first middleware computing system to a standard format of the middleware brokering server prior to being received by the middleware brokering server; and wherein the message is converted from the standard format to a native language format of the receiving second middleware computing system prior to being received by the receiving second middleware computing system." The Examiner suggested that adding these limitations would give a more complete picture of the claimed invention.

Furthermore, Claim 1 recites, "receiving the message sent from the first middleware computing system into a middleware brokering server." As established above, the middleware broker server of the present application allows computer systems operating under disparate types of middleware to communicate with each other in an efficient, verifiable, flexible, and maintainable manner. By contrast, Chiang et al. only teaches bi-directional data transformation between the client application and the server application using the Application metadata stored in metadata repository 709. (Col. 10, lines 45-65.) The novelty of Chiang et al. lies in the fact that it provides integration between the middleware and the enterprise applications without hard coded application program interfaces. Chiang et al. does not teach or suggest integration between one middleware service and another.

Claim 1 also recites, "sending the message from the middleware brokering server to a second middleware computing system that receives the message." The Office Action has suggested that IMS Connect is the second middleware computing system. However, as established above, IMS Connect is an adapter translating between XML documents and IMS message data structures. It is not a middleware service that allows multiple disparate applications to communicate with one another.

Furthermore, dependent Claims 7 and 14 depend directly from allowable independent Claim 1 and incorporate all of the limitations thereof. Accordingly, for the reasons established above, Applicants respectfully submit that Claims 1, 7, and 14 are not anticipated by Chiang et al. and respectfully request allowance of these claims.

Response to Rejections under Section 103

In the Final Office Action dated June 19, 2006, Claims 2-5, 8-10, and 17-21 were rejected under 35 USC § 103(a) as being unpatentable over Chiang et al. (U.S. Patent No. 6,948,174).

Dependent Claims 2-5, 8-10, and 17-21 depend directly or indirectly from allowable independent Claim 1 and incorporate all of the limitations thereof. Accordingly, for the reasons established above, Applicants respectfully submit that Claims 2-5, 8-10, and 17-21 are not obvious in view of the cited references and respectfully request allowance of these claims.

Conclusion

Applicants respectfully submit that the present application is in condition for allowance for the reasons stated above. If the Examiner has any questions or comments or otherwise feels it would be helpful in expediting the application, he is encouraged to telephone the undersigned at (972) 731-2288.

The Commissioner is hereby authorized to charge payment of any further fees associated with any of the foregoing papers submitted herewith, or to credit any overpayment thereof, to Deposit Account No. 21-0765, Sprint.

PAGE 13/14 * RCVD AT 8/7/2006 4:41:51 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-3/0 * DNIS:2738300 * CSID:9727312289 * DURATION (mm-ss):03-54

Respectfully submitted,

Date: August 7, 2006

Michael W. Piper
Reg. No. 39,800

ATTORNEY FOR APPLICANT

CONLEY ROSE, P.C.
5700 Granite Parkway, Suite 330
Plano, Texas 75024
(972) 731-2288
(972) 731-2289 (facsimile)